Practice 17: Conditional Operator and Recursion*

Kenneth Leroy Busbee

Abstract

Questions, exercises, problems, etc. that support this chapter in the "Programming Fundamentals - A Modular Structured Approach using C++" collection/textbook.

1 Learning Objectives

With 100% accuracy during a: memory building activity, exercises, lab assignment, problems, or timed quiz/exam; the student is expected to:

1. Define the terms on the definitions as listed in the modules associated with this chapter.
2. Understand the conditional operator and how it works.
3. Understand recursion as a problem solving technique.
4. When feasible, be able to convert C++ source code from a conditional expression to an if then else and vice versa.

2 Memory Building Activities

Link to: MBA 17[1]

3 Exercises

Exercise 1 *(Solution on p. 4.)*

Answer the following statements as either true or false:

1. The conditional expression acts like a case structure.
2. The conditional operator is a two part operator with three operands.
3. Recursion is one method of implementing a repetitive algorithm.
4. Recursion is always preferred over an iterative approach to a repetitive problem.
5. Factorial is usually demonstrated with an iterative approach.

*Version 1.6: Jan 13, 2010 2:21 pm -0600

†http://creativecommons.org/licenses/by/3.0/

[1]See the file at <http://cnx.org/content/m20815/latest/index.html>

4 Miscellaneous Items

None at this time.

5 Lab Assignment

5.1 Creating a Folder or Sub-Folder for Chapter 17 Files

Depending on your compiler/IDE, you should decide where to download and store source code files for processing. Prudence dictates that you create these folders as needed prior to downloading source code files. A suggested sub-folder for the **Bloodshed Dev-C++ 5 compiler/IDE** might be named:

- Chapter_17 within the folder named: Cpp_Source_Code_Files

If you have not done so, please create the folder(s) and/or sub-folder(s) as appropriate.

5.2 Download the Lab File(s)

Download and store the following file(s) to your storage device in the appropriate folder(s). You may need to right click on the link and select "Save Target As" in order to download the file.

Download from Connexions: Lab_17a.cpp[2]

5.3 Detailed Lab Instructions

Read and follow the directions below carefully, and perform the steps in the order listed.

- Compile and run the Lab_17a.cpp source code file. Understand how it works.
- Copy the source code file Lab_17a.cpp naming it: Lab_17b.cpp
- Convert the code that is using the if then else to a conditional expression.
- Convert the code that is using the conditional expression to an if then else.
- Build (compile and run) your program.
- After you have successfully written this program, if you are taking this course for college credit, follow the instructions from your professor/instructor for submitting it for grading.

6 Problems

6.1 Problem 17a – Instructions

Using proper C++ syntax, convert the following if then else to a conditional expression.

Example 1: if then else

```
if (x == y)
  {
  z = 14;
  }
else
  {
  z++;
  }
```

[2]See the file at <http://cnx.org/content/m20815/latest/Lab_17a.cpp>

6.2 Problem 17b – Instructions

Using proper C++ syntax, convert the following conditional expression to an if then else.

Example 2: conditional

```
answer = y < z ? 47 : 92;
```

Solutions to Exercises in this Module

Solution to Exercise (p. 1)

Answers:

1. false
2. true
3. true
4. false
5. false